

KA9Q Internet Protocol Package on the Apple Macintosh
by
Dewayne Hendricks WASDZP
Doug Thorn N6OYU

INTRODUCTION

The KA9Q Internet Protocol Package has been out for several years now and has made a decided impact on amateur packet radio. It has been implemented on several personal computer platforms, with the large majority of hams using it on IBM PC's and its clones. This article describes its implementation on the Apple Macintosh family of personal computers. The unique Macintosh user interface and its proper utilization by the KA9Q software has proved to be quite a challenge. We hope that users of our implementation will be pleased with the results.

USER INTERFACE DIFFERENCES

One of the major advantages the Macintosh provides to its users is the consistent user interface. This means the user generally has to learn only a few commands to become proficient with any program. However this requirement to support an 'ease of use' interface has caused some implementation difficulties, in our port of the KA9Q package to the Macintosh. One of the major issues that we had to overcome was the conversion of NET to a modeless program. In the Macintosh world all programs are modeless. This means that the user can perform any action at any time, regardless of the current state of the program.

For example, the MS-DOS implementation has a command line orientation. The user enters one command after another to cause the program to perform a given set of actions. In Macintosh programs the user can perform additional functions such as opening desk accessories (i.e. calculator, alarm clock, editor, etc.) together with the ability to cut and paste objects between programs at any time.

In our port to the Macintosh we have attempted to preserve the command line interface while allowing the traditional Macintosh functions which require a non-modal environment. This has resulted in an implementation which is familiar to the typical Macintosh user, but still preserves some of the look and feel of the original implementation.

FILE SYSTEM DIFFERENCES

One of the initial problems that we had to overcome in our port were the differences in the way the volume/directories are setup. In the MS-DOS world, you have disk drives, labeled A:, B:, C:, etc. Each volume can further be divided into multiple sub-directories, allowing you to logically place your files and documents in any location you wish. To specify the location of a document, you would use something like this:

On the Macintosh, the notation is quite different. An example best illustrates:

My Hard Disk:Amateur Radio:Packet files:KA9Q:NET/Mac

Note that the volume (and folders) may have spaces imbedded in the name, and that the names may be up to 15 characters long. Fortunately the only change we found necessary was to look for a ':' character instead of the '\' character when parsing the filenames and directories. When we need to specify a full path name that includes spaces, we just place the entire path in quotes. The implementation of these changes to the original package netted one additional feature; the ability to specify a different volume and path for each user when they access your station. This feature will be explained later in configuring the 'FTP Users' file.

APPLETALK SUPPORT

AppleTalk, which is the local area networking protocol built-in to every Macintosh, allowed us to provide several additional features. We decided to take advantage of this built-in capability by adding a driver to support the AppleTalk interface. Here's and example of how we were able to make use this support. One of us has in his QTH, a Macintosh Plus connected to a Yaseu FT-211 via an AEA PK-232. However, he does most of his work on a Macintosh 11x located across the room. Since AppleTalk is a networking protocol, all he had to do was connect the two computers together with a cable, and voila, he can now use his Mac 11x to operate on TCP/IP, **with no additional radio or TNC required**. The AppleTalk support allows him to assign another IP address to his Mac IIx, and send/receive files and mail via the Mac Plus to other stations. In fact, any number of Macintosh's can be connected (up to the limit of 254) to a single radio/TNC via the AppleTalk network. Appletalk also provides connection to LaserWriter printers that are shared on a local network. While this seems trivial, the issue is the limitation of two serial ports on the Macintosh, one used for the TNC and the other is used for network/printing.

MULTIFINDER OPERATION

MultiFinder, the pseudo-Multitasking environment of the Mac OS, allows us to run both Net/Mac and BM/Mac at the same time, each in it's own area of memory. This allows us to send and receive documents/mail while at the same time we are answering mail. The only requirement this has, is that we either need lots of memory (2.5 Megabytes suggested!), or are very careful in how the system files are setup. Although both Net/BM and Net/Mac have been tested on a Macintosh 512Ke, there just is not enough memory to run MultiFinder on that system, so we recommend at least a Macintosh Plus for this type of operation.

INSTALLATION

NET/Mac

Installation is fairly simple. Before getting started, the first thing you must do is to get an IP address (or several if you have more than one machine) assigned to you. You may wish to contact other TCP/IP users in your area to help in obtaining an address. In the meantime you will need to configure the system, connect your TNC to the Macintosh, and if you have a hard disk attached, copy the necessary files to your hard disk. We suggest you create a new folder somewhere on your hard disk, assuming you have one, if not then make a backup copy of the distribution diskette to work with. Now copy all the files and folders from the distribution diskette into this new folder on your working disk.

Directory structure

Make note of where you have placed all the files, Volume name, folder, etc., as you will need this information to configure several of the support documents.

Configuring the Autoexec.net

While there are a lot of commands that may be placed in the Autoexec.Net file, only a few are specific to the Macintosh implementation. The 'attach' command specifies to the program which serial port you wish to use to connect your TNC, the name you would like to call this port (ax0 in the distribution version), the packet size, and baud rates. An example best illustrates a typical attach command:

```
attach asy 0 A ax25 ax0 2048 256 9600
```

This command specifies the Modem port ('A'), packet sizes of 256/2048 and a baud rate of 9600.

Attach command for AppleTalk

The following line specifies the use of AppleTalk on the B port and a name of at0.

```
attach appletalk 77 B arpa at0 5 600
```

The above command specifies the printer port is to be used for Appletalk, and that up to 5 packets can be held on the receive queue. The size of the maximum transfer unit (MTU) is set at 600 bytes. The MTU size cannot exceed 600 bytes due to limitations in the AppleTalk protocol.

TZone

One other additional feature we added to Net/Mac project was the implementation of the **TZone** command. This command provides either local or GMT time stamping of mail. Below is an example of the command:

```
tzone PST 0
```

There are two parameters for this command, the first is your time zone, PST for Pacific Standard Time, or others as appropriate. The last parameter is the number of hours away from GMT you are located. For example here in California, we use -8 (subtract 8 hours from GMT to equal local time) when we wish to use GMT. If you wish to use local time, just leave this set to 0.

Use of the FTPUser's file as it relates to multiple volume support

The last file that needs to be edited, is the FTPUser's file. This plain text file lists all the user's, their password, and access level to the FTP server portion of the program. The only difference in the Macintosh version, is the support of foreign volumes. What this means is that we can now have more than one disk drive on-line, and point users to one of several volumes. As an example you could have a Macintosh running AppleShare™, the fileserver software from Apple Computer. Also running from the fileserver could be a land line BBS that shares it's files with the TCP/IP station. In this case the **FTPUsers** file would look like this:

```
Guest * FS:BBS:MacFiles 1  
WA8DZP Dewayne HD:PUB 7
```

The first entry allows anyone who logs into the TCP station with the name Guest, using any password, to have read only access to the BBS Files. The second entry allows WA8DZP, with the password of Dewayne to have read/write access a folder name PUB, on the hard disk (HD). Note that NET/Mac, like it's parent program for the MS-DOS world, allows access to all folders/files within the specified folder in the FTPUser's file. As a side note here... Be careful in how you construct your FTPUser's file. If you have two people both named Mike, with different passwords, then only the first Mike will have access to your system if you use Mike for the name, and their call signs for the password. It is suggested you use one's callsign as the name entry, and their name as the password.

BM/Mac

BM/Mac as the name may imply is the mail program for the Macintosh version of the KA9Q package. Installation is straight forward, with only the bm.rc file requiring editing.

Directory structure

BM/Mac requires several folders and files to be located in specific places in the system. First is the 'spool' folder which contains two additional folders, one named 'mqueue', and the other named 'mail'. The mqueue folder contains mail files that are queued up to be sent by the SMTP server within NET/Mac. Any mail that is being forwarded through your station will also be kept in the mqueue folder. The mail folder will contain mail that is received by the station.

BM.RC file

Configuration of the BM/Mac program requires editing the bm.rc file which must be located at the same folder level as the program. Below is a sample of the entries in the file:

```
host N6OYU
user Doug
fullname Douglas Thorn
reply n6oyu@n6oyu
zone PST
```

The first entry specifies the name of the host station. This entry **must** match the same entry in the Autoexec.Net file. The second 'user' entry is the default name of the user of the system. 'fullname' is obvious 'reply' is the name of the user and his host station name, which is used in the 'reply' field of the mail header. And finally zone is the time zone of the station.

Note: To be compatible with existing PBBS stations, use callsign@callsign in the 'reply' field if you plan on forwarding mail to the PBBS network

Alias file

This file is used by BM/Mac to allow the user to setup mail distribution lists or alias names for mail recipients. The mailer has been changed to first check the alias file for a name before using the information on a 'From:' or 'Reply to:' field when generating a return address. This is useful for having your own explicit return route to a mail recipient.

OPERATION

Multiple Session Window Support

One of the most useful additions to the Macintosh port has been multiple session windows. With this enhancement, every time the user creates a new session such as an FTP, a new window is created for that session where all of the input and output for the session will appear. The session which was most recently created will become the current active session. The user can switch

between sessions by either selecting the desired session from the 'Window' menu or by clicking upon the session actual window with the mouse.

The Console session window is the only session window that is active when the program is started. Other sessions are created in the normal manner from commands issued from the console session. The user can create a 'Log' session which shows the contents of the system log from the time the program was started. This can be scrolled by the user to see-what traffic the program has handled since it was started. One 'Trace' session can be started for any active interface. The trace window will show the output for the interface as specified by the user. Figure 1 below shows an example with three window open.

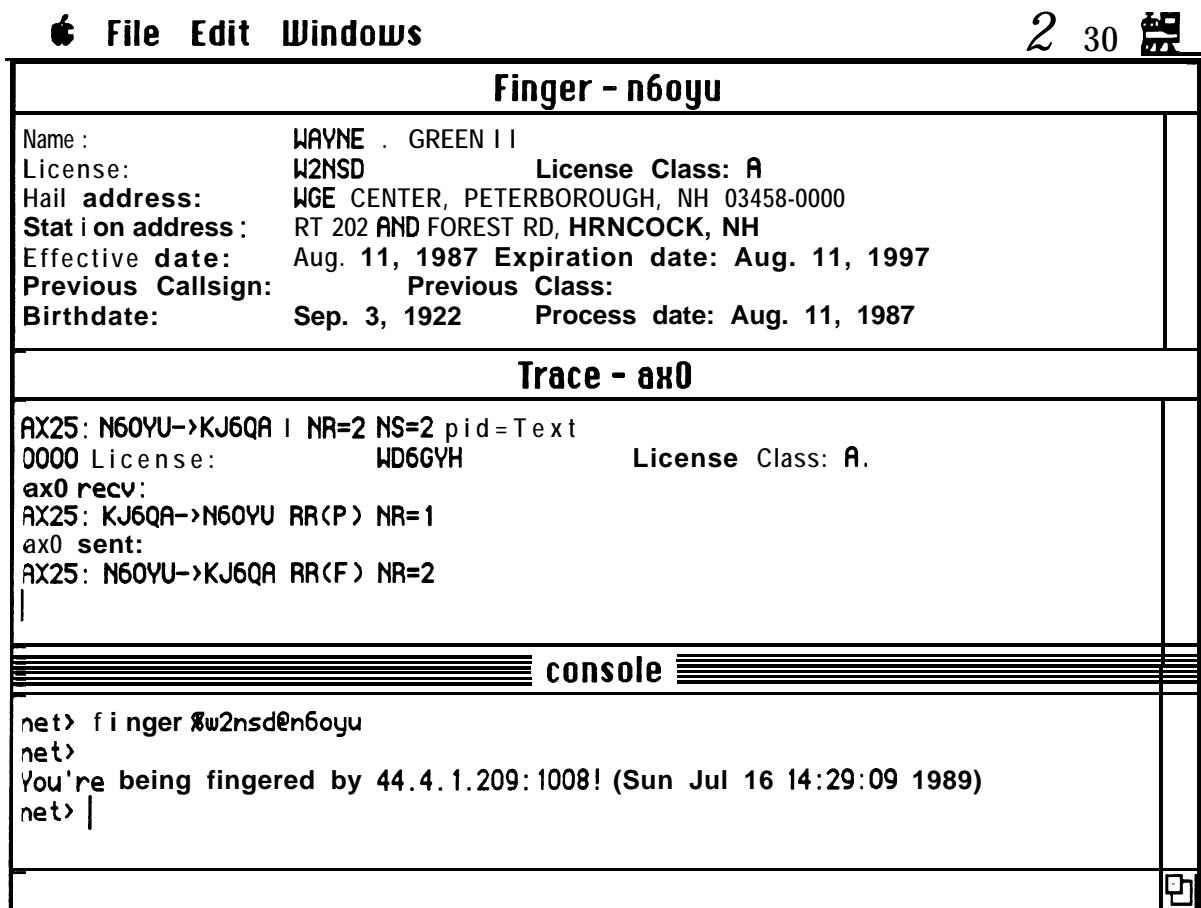


Figure 1

Finger session windows are handled in a special manner. They are allowed to stay open after the session has closed. This allows the user to have access to the information displayed in the window until it is no longer required. In addition, the 'Finger', 'Log' and 'Trace' session windows are treated as write-only. Since no input is allowed in these sessions, none is allowed in the window.

All session windows may be re-sized and placed on the screen in any way the user desires. This allows the activity on several sessions to be observed at the same time. This has proven to be a very useful feature during normal program operation.

MACINTOSH SPECIFIC ENHANCEMENTS

Help Facility

Another useful addition has been the on-line Help Facility to both NET/Mac and BM/Mac. The facility is activated when the user selects it from the 'Apple' menu and provides documentation on all of the commands available in each program. Examples are also provided for all of the commands, so that the user has some idea has to how each one should be used. Normal program operation continues while the Help Facility is active and when the user is done they can remove it by clicking on its window's close box. Figure 2 shows the initial Help display.

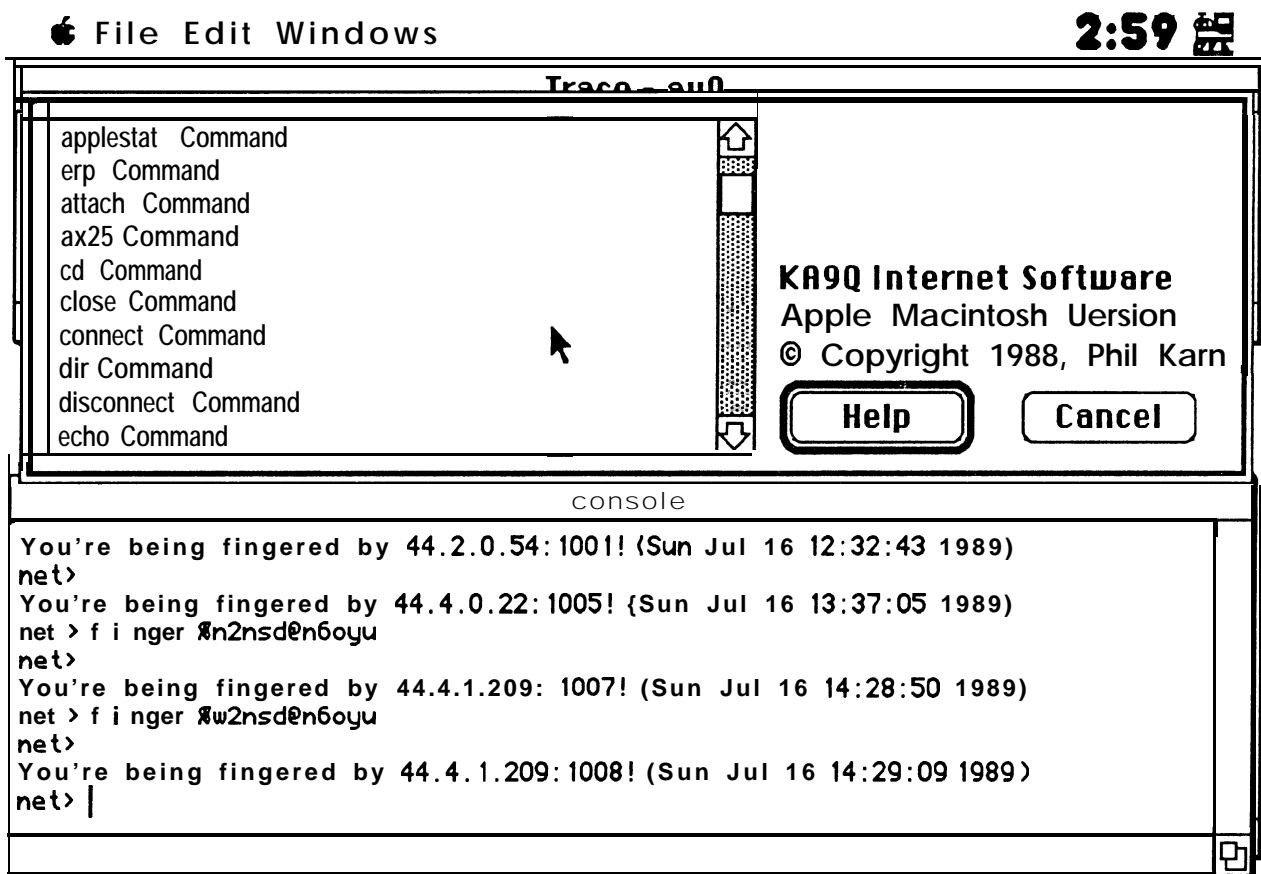


Figure 2

MacBinary II Support for FTP

In order to make it easier to transfer files between Macintosh systems running NET/Mac, the MacBinary II file transfer protocol has been added to the program. This protocol sends along with the normal data in a file, all of the Macintosh specific file information (i.e. program specific icons for the desktop). This is necessary as the Macintosh file system is quite different than that of MS-DOS systems and require additional information that is not transferred when an ‘image’ mode FTP transfer is done. A new sub-command “Macb” was added to the FTP client/server to provide this support. If a Macintosh user attempts to issue this command to an FTP server which is not on a Macintosh, then it is ignored.

AppleTalk Support

Support has been provided for communication to other Macintoshes running NET/Mac over an AppleTalk local area network by the addition of a Link Access Protocol (LAP) driver. With this driver, Macintoshes in the same zone can transfer ARP and IP packets to each other. The driver at this time does not allow communication across zones or internet operation.

FUTURE DIRECTIONS

NOS Implementation

In the near future, the next generation of the KA9Q Internet package, the Network Operating System (NOS) will replace the current version. At that time we expect to port all of our modifications and enhancements to that environment. As the architecture of the Macintosh is quite different from that of MS-DOS based systems, we expect that this will be a major undertaking which will require the direct replacement of many parts of NOS with Macintosh specific code. The Macintosh already has a multi-tasking environment called MultiFinder which provides many of the services implemented in the NOS kernel. We expect to make use of those services in our port of NOS to the Macintosh.

Inter-Application Interface

Apple recently announced a inter-application communication protocol (IAC) which will be available sometime in 1990. We expect to implement this protocol in the NOS version to allow Macintosh applications to inter-operate over a TCP/IP network. One application already planned for this interface is a callsign lookup server. We expect that this support will allow to have the callsign application access the callsign data in an SQL database running on a different host on the network.

Interoperation with other Macintosh TCP/IP Implementations

There are several other TCP/IP implementations available on the Macintosh. The most popular is from the National Center for Supercomputer Applications (NCSA) and is available in the public domain Apple has recently released their own version of TCP/IP, MacTCP which supports both AppleTalk and EtherTalk networks. We plan to provide support in NET/Mac to interoperate with both programs. This will greatly extend the utility of the program as a means of interconnecting different networking systems.

Virtual Terminal Support

With the Inter-Application interface support mentioned above, in order to provide a platform for the implementation of general applications we will need a generic terminal interface for the applications to talk to. We intend to implement a VT-100 style terminal session as an enhancement to the existing Telnet client/server. We had entertained the notion of providing only a Macintosh screen interface, but if we did this we would not be able to interoperate with applications on other non-Macintosh platforms. The VT-100 interface is commonly used in the industry and supports a superset of the ANSI standard escape sequences. We feel that this support will provide a good foundation for general application design.

CONCLUSION

The port of the KA9Q Internet package to the Macintosh has proven to be a very rewarding experience. The package on the Macintosh has quite a different look and feel than on some of the other platforms on which it is available. We hope to continue our efforts in improving the user interface of the package in order to provide the general packet community with an very easy to use, 'appliance-like' version of TCP/IP. We hope that our efforts will make TCP/IP available to more hams interested in exploring this new dimension in packet radio.